

Q-1) You are asked to code an application that monitors the performance data of athletes competing in different branches of athletics. For this purpose, you will need to create an Object Oriented hierarchy where **short distance runner** and **javelin thrower athletes** are modeled. Both of the models have common properties such as the name, surname and date of birth of the athletes. Furthermore, the both models include functions that can be invoked to print performance data of the athlete. You will need to create a function that prints the last N performance data of the athlete. You will also need to code a function that prints the best result of the athlete in his/her entire career. (Sizden atletlerin performansını farklı atletizm branşları için takip etmenizi sağlayan bir uygulama yazmanız isteniyor. Bu amaçla, kısa mesafe koşucusu ve mızrak atıcısı içeren nesne yönelimli hiyerarşiyi modellemeniz istenmektedir. İki atlet modeli de ad, soyad, doğum tarihi gibi ortak özelliklere sahiptir. Ek olarak, iki sınıfa ait model de atletlerin performans verilerini yazdıran iki adet fonksiyon içermelidir. Bu fonksiyonlar en son N adet performans verisini yazdıran ve en iyi sonucu yazdıran fonksiyonlardır.)

Each athlete shall have a dynamically created performance data record that shall keep data such as event time kept as string, the result as double(e.g. 155 meters for javelin throw), event city as string. Performance data of the athlete shall be set as the example code below.(Her atlet performans sonuçlarının saklandığı kayıtlardan oluşan ve dinamik olarak yaratılan bir dizi içermelidir. Performans verisi kaydında yarışma zamanını string, sonucu double (mesela 155 metre mızrak atışı) ve yarışma şehri string olarak tutulmalıdır. Performans verileri aşağıdaki verilen şekilde setlenebilir.)

The application code will need to have a function that prints the performance data of all the athletes that is sent to the function. An example function call can be made as following(uygulama kodu ayrıca atletlere ait performans verilerini ve en iyi zamanı yazdıran bir fonksiyonu main dosyasında gerçekleştirmelidir. Bu fonksiyon aşağıdaki gibi çağrılabilir):

```
JavelinThrower jt("Sedat", "Gormus", 15);
shortDistanceRunner sr("Mehmet", "Gormus", 15, 100);

jt.setPerformanceData("1/1/2018", 125, "Trabzon");
sr.setPerformanceData("1/1/2018", 9.95, "Trabzon");

printPerformanceData(jt);
printPerformanceData(sr);
```

a - 10 points) Create the class hierarchy including JavelinThrower and shortDistanceRunner with their public, protected and private data members and functions (Aşağıda verilen kutulara uygun şekilde sınıf hiyerarşisini oluşturun. Bu hiyerarşide mızrakAtıcısı ve kısaMesafeKoşucusu sınıfları olmalı ve bütün sınıfların public, protected ve private verileri ve metodları olmalıdır).



b - 15 points) Please create the listings for the classes (PS: you will need to write 3 .h files down, and you will need to understand the relationship between the classes)(Sınıfların listelerini veriniz (PS: 3 .h dosyası yazmalısınız ve sınıflar arasındaki ilişkileri düzgün şekilde belirlemelisiniz)).

C - 15 points) Please create the implementation files for the classes (PS: the files will need to implement constructors, destructors and all relevant functions implemented) (Sınıf gerçekleştirme dosyalarını (.CPP) oluşturunuz (PS: her .cpp dosyası constructors (yapıcı), destructors (yıkıcı) ve gerekli fonksiyonları içermelidir.).

d - 10 points) Please create the main function where you create a polymorphic calling of each classes print functions as in the example above. The print out of the functions should look like the following (main dosyasında olacak fonksiyonlarını gerçekleyiniz ve ilgili yazdırma fonksiyonunu polymorphic olarak yukarıdaki örnekte olduğu gibi çağırınız. Main dosyasının polymorphic çağırılması sonucundaki çıktı aşağıdaki gibi olmalıdır.).

Name : Sedat Surname : Gormus

1-) Date : 1/1/2018--Trabzon--125 meters for javelin throwing

Best Performance of Sedat Gormus is achieved on: 1/1/2018--Trabzon--125 meters

Name : Mehmet Surname : Gormus

1-) Date : 1/1/2018--Trabzon--9.95 seconds for 100 meters running

Best Performance of Mehmet Gormus is achieved on: 1/1/2018--Trabzon--9.95 seconds

e - 15 points) The JavelinThrower and shortDistanceRunner classes include a dynamically created array of performance data structures, can you please explain how to enable deep copy in these classes?. Furthermore, please overload the operator which will enable deep copying of the dynamically created data for only one class (either JavelinThrower or shortDistanceRunner)([mızrakAtıcısı ve kısaMesafeKoşucusu](#) sınıflar dinamik olarak oluşturulan ve performans verisi içeren dizilere sahiptir. Bu dizilerin derin kopyalamasının nasıl yapılacağını iki cümleyle açıklayınız ve derin kopyalama için gerekli operatörü sadece bir sınıf için tekrar tanımlayınız (operator overloading).).

f- 15 points) Can you please overload indexing operator ([]) for JavelinThrower class so that the performance data for the athlete can be accessed via the indexing operator. An example of how to use the indexing operator is given below. The indexing operator would return the performance data structure kept in the object. You have to make sure that the index is within the performance data array (Lütfen indexleme operatörünü ([]) mızrakAtıcısı sınıfı için tekrar tanımlayınız. Indexleme operatörü istenen indexteki performans verisini aşağıdaki gibi geri getirmelidir. Indexin dizinin sınırlarını aşmamasına dikkat edilmelidir.).

```
struct performance p = jt[1];
```

Q-2) Please briefly with **two sentences** explaining the following two OOP design principals (Lütfen aşağıdaki NYP kavramlarını **iki cümleyle** açıklayınız.).

a - 10 points) Encapsulation (Sarmalama)

b- 10 points) Friend functions (Arkadaş Fonksiyonlar)

Best of luck